

자바1

(연산자와 입출력)

Chapter 03

연산자와 입출력

오늘의 학습목표

- 연산자의 사용에 대해 이해한다.
- 변수와 연산자를 사용하여 연산을 진행한다.
- 각 연산자의 쓰임을 알고 %연산자를 사용할 수 있다.
- 기본 입출력문에 대해 공부한다.

연산자와 우선순위

연산자

- 특정한 연산을 수행하기 위해서 사용하는 기호
- 피연산자(operator) : 연산의 대상

종류	연산자	의미
산술	+, -, *, /, %	숫자 연산
대입	=, +=, -=, *=, **=, /=, %=	대입 및 복합 대입
관계	>, >=, <, <=, ==, !=	크기 비교
증감	++, --	증감 혹은 감소
논리	&&, , !	하나 이상의 논리적 처리
비트	&, , ^, ~, <<, >>	이진 연산
기타	논리식 ? 참 : 거짓	조건 연산

연산자와 우선순위

연산자 우선순위

- 연산자 우선순위가 확실하게 기억나지 않을 때에는 () 괄호를 사용하자

번호	결합규칙 (우선순위가 같은 경우에 식을 계산하는 방향)	연산자	의미
1	→(좌에서 우)	[값...], (값...)	자료형, 괄호
2	←(우에서 좌)	sizeof, &(주소), ++, --, ~!, +, -, 형변환	전위, 부호, 역참조 (단항 연산)
3	→(좌에서 우)	* / %	곱셈, 나눗셈, 나머지 (산술 연산)
4	→(좌에서 우)	+, -	덧셈, 뺄셈
5	→(좌에서 우)	>>, <<, >>>	비트 연산
6	→(좌에서 우)	<, <=, >=, >	관계 연산
7	→(좌에서 우)	==, !=	관계 연산
8	→(좌에서 우)	&	비트 연산
9	→(좌에서 우)	^	비트 연산
10	→(좌에서 우)		비트 연산
11	→(좌에서 우)	&&	논리연산
12	→(좌에서 우)		논리연산
13	←(우에서 좌)	?	3항 연산
14	←(우에서 좌)	=, +=, *=, /= 등	대입연산
15	→(좌에서 우)	,	콤마

연산자

1. 산술 연산

- 숫자 계산에 사용되는 연산자

연산자	의미	사용방법
+	덧셈	$c = a + b$
-	뺄셈	$c = a - b$
*	곱셈	$c = a * b$
/	나눗셈	$c = a / b$
%	나머지	$c = a \% b$

2. 대입 연산

- == 와 의미가 다름
- '변수 (연산=) 값'의 형태를 가짐
- 연산자를 붙여서 계산 후 대입

연산자	사용 방법
+=	$a += 2$
-=	$a -= 2$
*=	$a *= 2$
/=	$a /= 2$
%=	$a \% = 2$

3. 증감 연산

- 변수의 값을 증감시키거나 감소시킴

연산자	사용 방법	의미
++	$a++$	a를 증가시킨 후 연산
--	$a--$	a를 감소시킨 후 연산
++	$++a$	연산한 후 a를 증가시킴
--	$--a$	연산한 후 a를 감소시킴

연산자 예제1

1) 대입 연산자 예제

- 대입 연산자를 사용하여 연산 결과 값을 대입

```
public class Test {  
    public static void main(String[] args) {  
        int x = 1;  
        int y = 1;  
  
        x += 10;  
        y -= 15;  
  
        System.out.println(x);  
        System.out.println(y);  
    }  
}
```

2) 증감 연산자 예제

- 연산자 위치에 따라 출력되는 값이 달라짐

```
public class Test {  
    public static void main(String[] args) {  
  
        int x = 1;  
        int y = 1;  
  
        int nextx = ++x;  
        int nexty = y++;  
  
        System.out.println(nextx);  
        System.out.println(nexty);  
    }  
}
```

x를 증가시키고 nextx에 대입

nexty에 대입 후 y를 증가

연산자

4. 관계 연산

- 2개 항을 비교하여 논리 자료형으로 반환
- 결과는 True or False

연산자	사용법	의미	결과
>	a > 10	a는 10 초과	True or False
<	a < 10	a는 10 미만	
>=	a >= 10	a는 10 이상	
<=	a <= 10	a는 10 이하	
==	a == 10	a는 10과 같음	
!=	a != 10	a는 10이 아님	

```
public class Test {  
    public static void main(String[] args) {  
        int a = 15;  
  
        System.out.println(a > 10);  
        System.out.println(a < 10);  
        System.out.println(a == 10);  
        System.out.println(a == 15);  
        System.out.println(a != 10);  
    }  
}
```

연산자 예제2

윤년을 계산하는 프로그램을 작성

윤년 : 연도를 4로 나누어 딱 떨어지는 년

```
public class Test {  
    public static void main(String[] args) {  
  
        int year = 2012;  
        boolean isLeapYear;  
  
        isLeapYear = (year % 4 == 0);  
        System.out.println(isLeapYear);  
    }  
}
```


연산자

5. 논리 연산

- 2개 항을 논리적으로 연결할 때 사용
- 결과는 논리형 (True/False) 중 하나로 나옴

연산자	사용 방법	의미
&&	a && b	a와 b가 모두 참이면 True, 아니면 False
	a b	a와 b 중 하나라도 참이면 True, 아니면 False
!	!a	a가 참이면 False, a가 거짓이면 True

```
int a = 15;
int b = 10;
boolean c = true;

c = (a + b > 20) && (a - b < 20);
c = (a + b > 20) && (a - b > 20);

c = (a + b > 20) || (a - b < 20);
c = (a + b > 20) || (a - b > 20);

c = !(b < 10);
c = !(b <= 10);
```

연산자

6. 비트 연산

- 변수의 값을 0과 1의 조합인 2진수로 변환 후 연산을 수행
- AND, OR, NOT, XOR, SHIFT 연산

연산자	사용 방법	의미
&	a & b	a와 b를 비트 AND 처리
	a b	a와 b를 비트 OR 처리
^	a ^ b	a와 b를 비트 XOR 처리
~	~a	a를 비트 NOT 처리 (보수)
<<	a << 2	a의 비트를 왼쪽으로 2bit 이동
>>	a >> 2	a의 비트를 오른쪽으로 2bit 이동

AND : 입력이 모두 1이면 1, 아니면 0
OR : 입력이 하나라도 1이면 1, 아니면 0
XOR : 입력이 서로 다르면 1, 같으면 0

입력		연산결과			
p	q	p & q AND	p q OR	p ^ q XOR	~p NOT
0	0	0	0	0	1
0	1	0	1	1	1
1	0	0	1	1	0
1	1	1	1	0	0

연산자 실습

연산자 실습

변수 num1에는 150이, num2에는 240이 들어있다.

1. num1에 num2를 덧셈, 뺄셈, 곱셈, 나눗셈 한 결과를 오른쪽과 같이 출력하시오.

```
150 + 240 = 390
150 - 240 = -90
150 * 240 = 36000
150 / 240 = 0.625
```

2. num2의 3제곱의 결과에 num1를 나눈 몫이 10이상인지 True or False로 출력하시오.

결과 : True(혹은 False)

3. 3차 함수 $f(x) = 3x^3 - 6x^2 - 9x - 2$ 에서 함수 $f(2)$ 의 값을 구해서 출력하는 프로그램을 작성하시오.

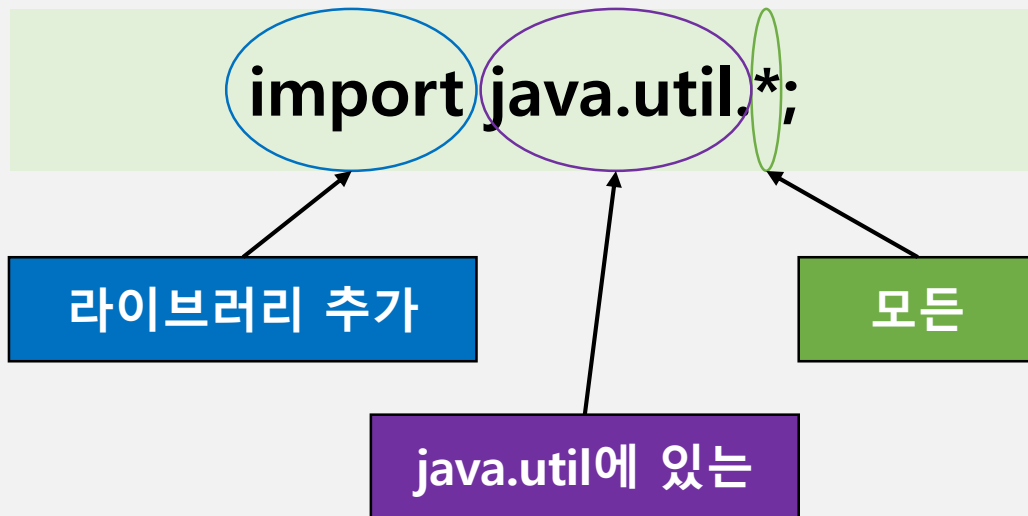
$f(2) = ?$

표준 입력

1. 입력문

- Scanner 클래스와 System.in을 사용
- Spacebar를 기준으로 문자열 파악
- EnterKey로 입력종료

1) import를 사용하여 Scanner 추가



2) Scanner 클래스에서 System.in기능 호출

```
Scanner input = new Scanner(System.in);
```

System.in을(입력기능) Scanner 클래스에 전달하여
input 변수에 저장

3) Scanner 클래스가 제공하는 다양한 기능을 사용하여 사용자로부터 입력받음

```
System.out.println("문장을 입력하세요 >>> ");  
String lineString = input.nextLine();
```

표준 입력

2. Scanner(System.in)의 입력 기능

번호	기능명	설명
1	next()	한 단어 (String)
2	nextLine()	한 줄 (String)
3	nextInt()	정수
4	nextDouble()	실수
5	nextFloat()	실수
6	nextLong()	정수
7	nextByte()	정수
8	nextShort()	정수
9	nextBoolean()	논리

표준 입력

3. 입력문 예

1) 한 줄 입력

```
System.out.println("문장을 입력하세요 >>> ");  
String line = input.nextLine();
```

2) 단어 입력

```
System.out.println("단어를 입력하세요 >>> ");  
String word = input.next();
```

3) 정수 입력

```
System.out.println("정수를 입력하세요 >>> ");  
int age = input.nextInt();
```

4) 실수 입력

```
System.out.println("실수를 입력하세요 >>> ");  
double divide = input.nextDouble();
```

5) 논리 입력

```
System.out.println("논리형을 입력하세요 >>> ");  
boolean check = input.nextBoolean();
```

표준 출력

1. 이스케이프 문자

- 확장 문자라는 의미로 / (back slash)(키보드 자판의 ₩)로 시작

문자	의미
₩'	작은 따옴표
₩"	큰따옴표
₩n	줄바꿈, LF
₩t	탭, TAB
₩₩	역슬래시, ₩₩

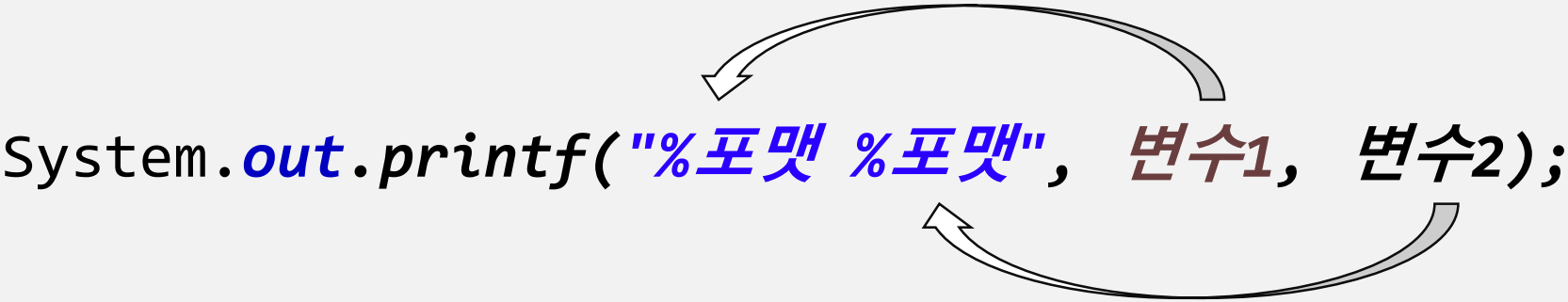
```
//큰따옴표
System.out.println("Hello World\\"!!!\\");
//작은따옴표
System.out.println("Hello World\'!!!\'");
//줄바꿈
System.out.println("Hello World\n!!!");
//Tab
System.out.println("Hello World\t!!!");
//역슬래시
System.out.println("Hello World\\!!!");
```

표준 출력

2. % 연산자

- 문자열 형식에 따라 사용하는 형식기호에 값을 전달하는 연산자
- System.out.printf() 메소드에서 사용 가능

구분	포맷표현	자료형
int	%d, %o, %x	정수
float	%f	실수
char	%c	문자형
String	%s	문자열
boolean	%b	논리형



표준 출력

2-1. % 연산자 예제

1) 기본 포맷 사용 (a와 b변수에 들어있는 값 출력)

```
int a = 10;  
int b = 20;  
  
System.out.printf("a = %d, b = %d", a, b);
```

2) 연산 사용 (a와 b변수에 들어있는 값을 이용해 연산 결과 출력)

```
int a = 10;  
int b = 20;  
  
System.out.printf("%d + %d = %d", a, b, a+b);
```

3) 자료형 포맷 사용 (a와 b변수 자료형을 변경, 칸 설정)

```
int a = 10;  
int b = 20;  
  
System.out.printf("a = %6.2f, b= %5d", (float)a, b);
```

%6.2f의 의미

- 실수를 6개의 칸에 표시
- 소수점 이하는 2자리



```
System.out.printf("%포맷 %포맷", 변수1, 변수2);
```

Chapter3. 연산자와 입출력

입출력 실습

1. 반지름을 입력 받아서 원의 둘레와 원의 면적을 계산해서 출력하는 프로그램을 구현하세요.

$\text{원의 둘레} = \text{지름} * 2$

$\text{원의 면적} = \text{반지름} * \text{반지름} * 3.14(\text{파이})$

반지름을 입력하세요 >>> 3

원의 둘레 = 12

원의 면적 = 28.26

2. 직사각형의 가로와 세로 값을 입력으로 받아서 직사각형의 둘레와 넓이를 구하는 프로그램을 구현

가로 길이를 입력하세요 >>> 3

세로 길이를 입력하세요 >>> 5

직사각형의 둘레 = 16

직사각형의 넓이 = 15

3. 숫자를 입력 받아서 해당 숫자가 짝수면 true, 홀수면 false를 출력하는 프로그램을 구현하세요.

숫자를 입력하세요 >>> 6

6은 true입니다.

숫자를 입력하세요 >>> 25

25은 false입니다.

수고하셨습니다.